PLEASE AMEND THE CLAIMS AS FOLLOW:

1. (Currently Amended) A computing system, the system comprising:

a network coupling a first computing subsystem and a second computing subsystem;

wherein the second computing subsystem provides sending of streaming data packets containing digital media to the first computing subsystem;

wherein the first computing subsystem provides means for:

(a) receiving of the streaming data packets containing digital media from the second computing subsystem,

(b) utilizing an operational software module by the first computing subsystem for processing of the streaming data packets containing digital media ~~to provide at least one of audio playing on a sound device and a video playing on a display device~~, and

(c) utilizing the operational software module by the first computing subsystem for generating ~~of~~ security tags responsive to said processing of streaming data packets and sending the security tags to the second subsystem; and

wherein the second computing subsystem provides means for:

(a) receiving the security tags from the first computing subsystem, and

(b) providing processing logic for validating as a successful validation that the operational software module was unchanged when utilized in generating the ~~received~~ security tags ~~from~~ at the first computing subsystem, and otherwise determining a failed validation if the operational software module was changed when utilized in generating the security tags at the first computing subsystem.

~~wherein a successful validation determines that the respective security tags were properly generated responsive to said processing of streaming data packets, and~~

~~wherein a failed validation determines that the respective security tags were improperly generated responsive to said processing of streaming data packets.~~

2. (Currently Amended) The system as in Claim 1, wherein the second computing subsystem further comprises a transmission and forwarding controller responsive to the processing logic for validating, for performing at least one of the following:

(a) stopping the sending of the streaming data packets containing digital media to the first computing subsystem responsive to the determining~~ation~~ of the respective failed validation~~; and~~

~~(b) stopping the forwarding of further streaming data packets containing digital media to the first computing subsystem responsive to determination of the respective failed validation.~~

3. (Previously Presented) The system as in Claim 1, wherein the operational software module provides rules of processing that are defined by at least one of: a content management subsystem, a digital right management subsystem, and predefined policy rules associated with the content.

4. (Previously Presented) The system as in Claim 1, further comprising: a defined sequence of decryption keys; and

wherein portions of the defined sequence of decryption keys are sent, one portion at a time, from the second computing subsystem to first computing subsystem, responsive to successful validation of the processing logic for validating the received security tags from the first computing subsystem.

5. (Currently Amended) The system as in Claim 1, wherein the processing of streaming data packets on the first computing subsystem is further responsive to a at least one of: a privileges table, a privileges decision-tree, pseudo random rendering logic, a streaming data

packet header processing privileges decision-tree, a security tag processing logic, a streaming data packet identification processing logic, a secure time-stamp processing logic, a time stamp representing execution time of at least one selected operation, a processing of streaming data packets with secure time-stamps, watermarking information processing, fingerprinting information processing, stenographic information processing, data embedding information processing, digital signature information processing, and a processing of streaming data packets with secure time-stamps that is responsive to UTC (coordinated universal time).

6. (Previously Presented) The system as in Claim 1, wherein the processing of streaming data packets on the first computing subsystem utilizes codes and parameters defining and expressing at least one of: privileges, authorizations, access rights, and entitlements as expressed in XrML (Extensible Rights Markup Language).

7. (Currently Amended) The system as in Claim 1, wherein at least one of: the processing of streaming data packets and the generating of security tags are further responsive to at least one of:

a predefined schedule, a secure time-stamp, a time stamps representing execution time of at least one selected operation, renewable codes and parameters, updated codes and parameters, a predefined schedule received from the second computing subsystem, a secure time-stamp received from the second computing subsystem, renewable codes and parameters received from the second computing subsystem, updated codes and parameters received from the second computing subsystem, replacement codes and parameters received from the second computing subsystem, a predefined schedule received from a third computing subsystem, a secure time-stamp received from a third computing subsystem, renewable codes and parameters received from a third computing subsystem, and updated codes and parameters received from a third computing subsystem.

8. (Previously Presented) The system as in Claim 7, wherein at least one of: selected parts of the processing of the streaming data packets, selected parts of the generating of security tags, selected parts of the renewable codes and parameters, and selected parts of the updated codes and parameters, are provided from an external storage medium.

9. (Previously Presented) The system as in Claim 8, wherein the external storage medium is at least one of: a smart card, a tamper-resistant device, an obfuscated storage, a hidden storage, an encrypted data storage, a removable storage device , a token card, a network interface, a wireless access point, a wireless base station, and a metro card.

10. (Previously Presented) The system as in Claim 1, wherein selected parts of the processing of streaming data packets, and selected parts of the generating of security tags are defined as a plurality of logic modules that are interlocked to provide concurrent execution.

11. (Previously Presented) The system as in Claim 1, wherein the display device is at least one of:

an analog display, a digital display, a television, a flat panel display, a screen, a movie screen, a liquid crystal display, a solid state display, a home video system, a computer display, a CRT display, a mobile phone display, a PDA display, a three-dimensional display, a holographic display, a computer monitor, a handheld display, a digital output system, an electronic book display, and an analog output system.

12. (Previously Presented) The system as in Claim 1, further comprising: an update controller separate from the first computing subsystem, and providing at least one of: updated codes, updated parameters, update decryption codes, update decryption keys, update rendering codes, update playing codes, and updated secure time stamp to the first subsystem.

13. (Previously Presented) The system as in Claim 12, further comprising: a security management server (SMS) for providing update information to the update controller.

14. (Currently Amended) The system as in Claim 1,
wherein the utilizing an operational software module by the first computing subsystem for processing of the streaming data packets containing digital media provides at least one of audio playing on a sound device and a video playing on a display device,
wherein the sound device is at least one of: a speaker, a plurality of speakers, a surrounding sound system, ear phones, loudspeakers, a high fidelity sound system, a stereo audio system, a digital output system, and an analog output system.

15. (Canceled)

16. (Previously Presented) The system as in Claim 1, wherein the first computing subsystem is further comprised of cryptographic modules; and

wherein the cryptographic modules provide for at least one of:

program authentication, user authentication, cryptographic authentication, application authentication, encryption, a secure time-stamp, a digital signature, watermarking information, IPSec (IP Security) functionality, TLS (Transport Layer Security) functionality, and SSL (Secure Sockets Layer) functionality.

17. (Previously Presented) The system as in Claim 1, wherein the second computing subsystem is further comprised of validation modules; and

wherein the validation modules further provide at least one of:

program authentication, user authentication, cryptographic authentication, application authentication, encryption, a secure time-stamp, a digital signature, watermarking information, IPSec (IP Security) functionality, TLS (Transport Layer Security) functionality, and SSL (Secure Sockets Layer) functionality.

18. (Canceled)

19. (Canceled)

20. (Canceled)

21. (Currently Amended) The system as in Claim 1,

wherein the utilizing an operational software module by the first computing subsystem for processing of the streaming data packets containing digital media provides at least one of audio playing on a sound device and a video playing on a display device, and

wherein the processing of the streaming data packets provides at least one of: the audio playing and the video playing, and further provides at least one of:

deleting streaming data packets after processing, deleting streaming data packets within a predefined time interval after processing, deleting streaming data packets after a defined number of times of processing, preventing copying of the streaming data packets, preventing printing of the streaming data packets, preventing sending of the streaming data packets, encrypting video rendering of content received in the streaming data packets, pseudo random video rendering of content received in the streaming data packets, and encrypted video rendering of content received in the streaming data packets.

22. (Currently Amended) The system as in Claim 1,

wherein the utilizing an operational software module by the first computing subsystem for processing of the streaming data packets containing digital media provides at least one of audio playing on a sound device and a video playing on a display device, and

wherein at least one of: the audio playing and the video playing operate in accordance with logic based upon at least one of:

XrML (Extensible Rights Markup Language) specifications, trusted computing specifications, trusted computing based principles, validation of watermarking information, IPSec (IP Security) functionality, TLS (Transport Layer Security) functionality, and SSL (Secure Sockets Layer) functionality.

23. (Previously Presented) The system as in Claim 1,
wherein the second computing subsystem further includes a media server.

24. (Previously Presented) The system as in Claim 23, wherein there is a plurality of the first computing subsystems, each coupled to the network and receiving streaming data packets from the second computing subsystem, and wherein the media server regulates distribution of the streaming data packets to only a predefined number of the plurality of the first computing subsystems.

25. (Previously Presented) The system as in Claim 1, wherein there is a plurality of the first computing subsystems, each coupled to the network and receiving streaming data packets from the second computing subsystem.

26. (Previously Presented) The system as in Claim 25, wherein the second computing subsystem encodes the respective streaming data packets responsive to the respective successful validation of the respective received security tags from the respective one of the plurality of the first computing subsystems.

27. (Previously Presented) The system as in Claim 26, wherein the streaming data packets are encoded such that any of the plurality of the first computing subsystems in which the validating of their security tags fail will not thereafter be able to further decode the streaming data packets.

28. (Previously Presented) The system as in Claim 25, wherein the streaming data packets are encrypted by the second computing subsystem by using a group encryption scheme such that any of the first computing subsystems in which the validating of their security tags fail will not be able to further decode the streaming data packets.

29. (Previously Presented) The system as in Claim 26, wherein the streaming data packets are sent using at least one of:

multicast, IP (Internet Protocol) multicast, Secure IP multicast, group key management architecture, multi-party non-repudiation protocol, group communications, and secure group communications.

30. (Previously Presented) The system as in Claim 1, wherein there is a plurality of second computing subsystems coupled to the network, each sending a respective plurality of streaming data packets to the first computing subsystem.

31. (Previously Presented) The system as in Claim 30, wherein the first computing subsystem sends security tags to the plurality of second computing subsystems for validation.

32. (Previously Presented) The system as in Claim 1, wherein the first computing subsystem is at least one of:

a computer, a wireless device, a handheld device, a Wi-Fi device, a device operating in accordance with IEEE 802.11 family of standards, a device operating in accordance with IEEE 802.15, a device operating in accordance with IEEE 802.16, a 2.5G cellular telephone, a 3G cellular telephone, a 4G cellular telephone, a 5G cellular telephone, a personal computer, a set-top box, a device operating in accordance with UMTS (universal mobile telephone system), and a device operating in accordance to the IEEE 802 family of standards.

33. (Previously Presented) The system as in Claim 1, wherein the second computing subsystem further comprises encryption logic for encrypting streaming data packets prior to sending the respective streaming data packets; and

wherein the first computing subsystem processing of streaming data packets further comprises logic for decrypting the streaming data packets.

34. (Previously Presented) The system as in Claim 33, wherein the second computing subsystem provides a decryption key to the first computing subsystem processing logic.

35. (Previously Presented) The system as Claim 34, wherein the encryption key is provided for at least one of: periodically, at random times, at predefined time intervals, responsive to validating the security tags, and at predefined times derived from coordinated universal time (UTC).

36. (Previously Presented) The system as in Claim 33, wherein the first computing subsystem further comprises logic for generating and sending encryption keys to the second computing subsystem from at least one of the following: a smart card external device and a trusted platform module (TPM); and

wherein the second computing subsystem uses the encryption keys for encrypting the streaming data packets prior to sending them.

37. (Currently Amended) A method for authenticating, at a second computing subsystem, a software module utilized in operation ~~between~~ at ~~least~~ a first computing ~~element~~subsystem ~~and a second computing element~~, the method comprising:

sending of data to the first computing ~~element~~ subsystem from the second computing ~~element~~subsystem;

receiving the data in the first computing ~~element~~subsystem;

processing the data on the first computing ~~element~~subsystem responsive to an operational ~~logic~~software module, and generating security tags responsive to an associated tag generation module concurrently executing with the operational software module;

~~wherein~~ concurrently executing the associated tag generation module ~~is only executable concurrently with~~ responsive only to executing the respective operational ~~logic~~software module in the first computing ~~element~~subsystem;

~~wherein the processing provides playing that provides at leas one of: an input to an audio device and an input to a video display;~~

sending the security tags to the second computing ~~element~~subsystem from the first computing ~~element~~subsystem;

processing the security tags in the second computing ~~element~~subsystem to determine ~~one of:~~successful validation responsive to validating that the processing of the data in the first computing ~~element~~subsystem was processed by the operational ~~logie~~software module operating concurrently with the generating the security tags by the associated tag generation, and otherwise determining a failed validation ~~responsive to validating that the processing of the data in the first computing element was not processed by the operational logic module operating concurrently with the generating the security tags by the associated tag generation~~; and

adjusting communication of sending of the data responsive to one of the successful validation and the failed validation.~~stopping sending further data to the first computing element responsive to the failed validation.~~

38. (Currently Amended) The method as in Claim 37, further comprising:

providing a plurality of operational software ~~logie~~ modules and parameters operable stand-alone to provide a respective plurality of subtask functions

providing secure integration of the plurality of operational software ~~logie~~ modules and parameters to provide a combined functionality;

interlocking the plurality of operational software ~~logie~~ modules and parameters into a single logic program that is only operable to concurrently execute the plurality of operational software ~~logie~~ modules; and

providing the combined functionality responsive to the plurality of subtask functions concurrently executing responsive to the single logic program at the first computing element.

39. (Currently Amended) The method as in Claim 37, wherein the generating security tags is at least one of: forms a pseudo-random sequence of security tags and a time stamp representing execution time of at least one selected operations.

40. (Previously Presented) The method as in Claim 39, further comprising:

producing the pseudo-random sequence of security tags utilizing computation according to at least one of:

applying a pseudo-random generator, applying a pseudo-random function, applying a cryptographic function, applying an encryption function, applying a scrambling subroutine, applying an authentication function, applying a digital signing function, applying a cryptographic hash function, applying a subroutine, applying a computational logic module, applying a symmetric cryptography function, applying an asymmetric cryptography function, employing a cryptographic key, employing a cryptographic seed, employing an encrypted software, employing an obfuscated software, employing a hidden program, employing watermarking information, employing fingerprinting information, employing digital signature information, employing logic with a set of parameters, employing a hardware module, employing a trusted platform module (TPM), employing a smart card, employing a portable device, and employing a distributed protocol.

41. (Canceled)

42. (Canceled)

43. (Previously Presented) The method as in Claim 38, further comprising:

an external smart card device that is part of the first computing element, and

wherein selected modules and parameters of the plurality of software logic modules and parameters reside on the external smart card device.

44. (Original) The method as in Claim 38, wherein selected ones of the parameters are at least one of: an encryption key, a decryption key, and an authentication parameter.

45. (Currently Amended) The method as in Claim 38, further comprising:

renewing selected ones of the plurality of operational software ~~logic~~ modules and parameters.

46. (Currently Amended) The method as in Claim 38, further comprising:

replacing selected ones of the plurality of operational software ~~logic~~ modules and parameters.

47. (Previously Presented) The method as in Claim 45, wherein the renewing is performed in at least one of:

periodically, at random times, at predefined times, at predefined times derived from coordinated universal time (UTC), responsive to receiving data by the first computing element, responsive to sending the security tags, and responsive to sending data by the second computing element.

48. (Previously Presented) The method as in Claim 38, wherein the data is stored in a memory in the first computing element responsive to the receiving the data in the first computing element; the method further comprising:

erasing the data from the memory in the first computing element by the single logic program in the first computing element.

49. (Previously Presented) The method as in Claim 48, wherein the memory is at least one of: a solid state device, random access memory, a main memory, a secondary memory, a magnetic storage device, and an optical storage device.

50. (Previously Presented) The method as in Claim 48, wherein the erasing the data is performed at least one of:

after a predefined time, after the data has been output to an output device, and after the data has been output a predefined number of times to an output device.

51. (Previously Presented) The method as in Claim 37, wherein the first computing element is comprised of at least one of:

a computer, a wireless device, a handheld device, a Wi-Fi device, a device operating in accordance with IEEE 802.11, a device operating in accordance with IEEE 802.15, a device operating in accordance with IEEE 802.16, 2.5G cellular telephone, a 3G cellular telephone, a 4G cellular telephone, a 5G cellular telephone, a personal computer, a computing subsystem, a set-top box, a device operating in accordance with UMTS (Universal Mobile Telephone System), and a device operating in accordance with IEEE 802.3 family of standards.

52. (Currently Amended) A method of ~~providing content~~ protected~~ion~~ ~~in~~ communicating of streaming data packets, the method comprising:

~~defining within a first computing subsystem, defined rules for processing;~~

receiving the streaming data packets containing content, in the first computing subsystem;

processing of the streaming data packets in the first computing subsystem according to defined rules for processing;

generating security tags responsive to execution of the defined rules for processing;

sending the security tags from the first computing subsystem to a second computing subsystem;

providing security ~~defined~~ tag validation logic in the second computing subsystem ~~relating to said generating of security tags~~;

processing, in the second computing subsystem, the received security tags, responsive to the security ~~defined~~ tag validation logic to provide respective validated security tags; and

processing in the second computing subsystem the validated security tags and the received security tags to determine whether the generating security tags in the first computing subsystem was properly generated responsive to execution of the defined rules for processing at the first computing subsystem so as to validate that the defined rules of processing were unchanged at the time of execution at the first computing subsystem.

53. (Previously Presented) The method as in claim 52, wherein the content is representative of at least one of: a movie, a book, a music piece, a concert, a 3D movie, a sport event, a text file, and a multimedia file;

wherein the content is divided into predefined number of parts, and

wherein each of the parts is associated with a decryption key.

54. (Currently Amended) A method for authentication of integrity of software executed in ~~generation~~ generating of communicated data packets within a computer, the method comprising:

transmitting data packets from a second computing subsystem to a first computing subsystem;

receiving the streaming data packets for processing in the first computing subsystem;

providing operation software that provides for defining rules of processing for execution on the first computer subsystem;

~~wherein the processing provides playing that provides at least one of: an input to an audio device and an input to a video display;~~

generating security tags at the first computing subsystem responsive to the streaming data packets and responsive to the rules of processing;

sending respective ones of the security tags from the first computing subsystem to the second computing subsystem~~, responsive to the streaming data packets and the rules of processing~~; and

processing the received security tags in the second computing subsystem to validate that the operation software was unchanged when the operation software performed the processing in the first computing subsystem ~~is~~ when operating according to the rules of processing.

55. (Currently Amended) The method as in Claim 54, further comprising:

validating the received security tags in the second computing subsystem responsive to second rules of processing; and

processing the received security tags in the second computing subsystem to validate execution of ~~a correct logic module~~the operation software providing defining of the rules of processing in the first computing subsystem, to validate that the processing of the streaming data packets in the first computing subsystem is according to the respective defined rules for processing utilizing the respective ~~correct logic module~~unchanged operation software.

56. (Currently Amended) A computer system providing remote authentication on a first computing subsystem of processing of content at a remote computing subsystem, the system comprising:

a tag generator at the remote computing subsystem operating from an initial generator state to locally generate a sequence of security tags responsive to concurrent execution ~~upon~~ of an operational code module utilizing a sequence of content processing steps;

means providing for transmission from the remote computing subsystem to the first computing subsystem of the sequence of security tags;

a tag verifier at the first computing subsystem, operating from an initial verification state to generate a sequence of comparison security tags for selective comparison to the sequence of the security tags; and

means for coordinating the initial generator state and the initial verifier state prior to the execution ~~on the sequence of content processing steps~~of the operational code module,

wherein the tag verifier selectively provides valid comparison tags responsive to the means for coordinating, wherein the valid comparison tags are utilized to ~~provide authentication of~~authenticate that the operation code module was unchanged during the execution at the remote computing subsystem ~~of~~ for assuring integrity of the sequence of content processing steps.

57. (Previously Presented) The system as in Claim 56, wherein the tag generator includes a sequence number as part of the security tags.

58. (Previously Presented) The system as in Claim 57, wherein the tag verifier generates a comparison sequence number for selective comparison to the sequence number that is part of the security tags.

59. (Previously Presented) The system as in Claim 57, wherein the sequence number is used for at least detecting a loss of a respective security tag.

60. (Previously Presented) The system as in Claim 56, wherein the tag generator provides a secure time-stamp as part of each of the security tags.

61. (Previously Presented) The system as in Claim 60, wherein the tag verifier generates a comparison secure time-stamp for selective comparison to the secure time-stamp that is part of each of the security tags.

62. (Previously Presented) The system as in Claim 56, further comprising:

means for remotely downloading of codes and parameters for use with the tag generator and the sequence of content processing steps.

63. (Previously Presented) The system as in Claim 62, wherein the codes and parameters are used to perform at least one of: processing to provide the sequence of content processing steps, and generating security tags.

64. (Currently Amended) A system for authenticating operation by providing secure integration of separate software logic modules to provide a combined functionality, the system comprising:

~~a first computing subsystem, wherein the first computing subsystem provides:~~

~~(a) receiving of streaming data packets from a media server associated with the first computing subsystem, and providing processing of the streaming data packets, responsive to defined rules for processing of the streaming data packets, and~~

~~(b) generation of security tags and selectively sending of the security tags to said media server associated with the first computing subsystem;~~

~~the system further comprising:~~

a plurality of software logic modules each operable stand-alone to provide a respective one of a plurality of subtask functions associated with operations on ~~the~~ a first computing subsystem; and

a transformation controller providing interlocking of the plurality of software logic modules, comprised at least of a first separate operational module and a second separate operational module, into a single logic program that provides a combined functionality;

wherein the combined functionality is only provided by the first computing subsystem when the plurality of subtask functions are executed concurrently responsive to the single logic program;

wherein the combined functionality provides for the first computing subsystem providing the subtask functions of:

(a) receiving of streaming data packets from a media server associated with the first computing subsystem responsive to the first separate operational module within the single logic program, and providing processing of the streaming data packets responsive to defined rules for processing of the streaming data packets, and

(b) generation of security tags responsive to the second separate operational module within the single logic program and selectively sending of the security tags to said media server associated with the first computing subsystem.

wherein the combined functionality assures that the first separate operational software module for receiving of data packets executes concurrently with the second separate software logic module generating security tags.

65. (Previously Presented) The system as in Claim 64, wherein the single logic program is written to be immune to reverse generation.

66. (Currently Amended) The system as in Claim 64, wherein one of the software logic modules provides a function for producing at least one of: pseudo-random sequence of security tags and time stamps representing the execution time of selected operations, at the first computing subsystem;

wherein at least one of: the pseudo-random sequence of security tags and the time stamps representing the execution time of selected operations, are sent to the second computing subsystem for verification of correct processing on the first computing subsystem; and

wherein the streaming data packets are sent to the first computing subsystem from said media server, only upon successful verification that the combined functionality was executed on the first computing subsystem.

67. (Currently Amended) The system as in Claim 66, wherein at least one of producing: the pseudo-random sequence of security tags and the time stamps representing the execution time of selected operations, utilizes computation by at least one of:

applying a pseudo-random generator, applying a pseudo-random function, applying a cryptographic function, applying an encryption function, applying a scrambling subroutine, applying an authentication function, applying a digital signing function, applying a cryptographic hash function, applying a subroutine, applying a computational logic module, applying a symmetric cryptography function, applying an asymmetric cryptography function, employing a cryptographic key, employing a cryptographic seed, employing an encrypted software, employing an obfuscated software, employing a hidden program, employing logic with a set of parameters, employing a hardware module, employing a smart card, employing a portable device, employing local clock, employing universal time, and employing a distributed protocol.

68. (Previously Presented) The system as in Claim 64, wherein one of the software logic modules provides logic to process content of the streaming data packets.

69. (Previously Presented) The system as in Claim 68, wherein logic to process the content of the streaming data packets performs at least one of:

video rendering of the content on a video display, playing the content via audio speakers, displaying the content on an e-book output device, outputting the content to an output device, and outputting the content to an analog output device.

70. (Previously Presented) The system as in Claim 64, wherein one of the software logic modules provides rules of playing of audio and video content.

71. (Previously Presented) The system as in Claim 70, wherein the rules of playing of audio and video content ensure at least one of:

the content is not printed; the content is not sent to a third party; the content is destroyed after being displayed on a video monitor; the content is being destroyed after being played via an audio speakers; the content is erased from all memory storage devices after being displayed on a video monitor; the content is erased from all memory storage devices after being played via an audio speakers; the content is erased from all memory storage devices after being used via an e-book output device; the content is erased from all memory storage devices after a predefined time interval; the content is erased from all memory storage devices at a time defined time by coordinated universal time (UTC); the content is used in accordance with rights defined using XrML (Extensible Rights Markup Language) specifications; the content is used in accordance with trusted computing specifications; the content is used in accordance with trusted computing based principles; and the content is used in accordance with at least one of the following: watermarking information, stenographic information, fingerprinting information, embedded data and digital signature information.

72. (Previously Presented) The system as in Claim 70, wherein the rules of playing provide at least one of: content processing and determining a renewable software for content processing.

73. (Previously Presented) The system as in Claim 72, wherein the renewable software for content processing is at least one of:

a number of times the content can be displayed, a number of times the content can be played, a time signal, a UTC time signal, a digitally signed time signal, a software element, a predefined task, a code for processing content signature, and a code for watermarking the content.

74. (Previously Presented) The system as in Claim 72, wherein the renewable software for content processing is obtained from at least one of:

a second computing subsystem, a second computing element, predefined logic, an external rule controller, a security management system, via a network interface, a

network appliance, a server, a network management system, a firewall, a local computing subsystem, a smart card device, a trusted platform module (TPM) device, and a portable device.

75. (Previously Presented) The system as in Claim 64, wherein one of the software logic modules provides a cryptographic function for producing a pseudo-random sequence of security tags; and

wherein one of the software logic modules provides logic to process and play audio and video content.

76. (Original) The system as in Claim 64, wherein one of the software logic modules provides a cryptographic function for verifying at least one of: watermarking, embedded data and fingerprinting.

77. (Previously Presented) The system as in Claim 64, further comprising:

a source of interlocking parameters, and

wherein the transformation controller is further comprised of means for combining the software logic modules responsive to the interlocking parameters.

78. (Original) The system as in Claim 77, wherein the source of interlocking parameters is generated by at least one of:

a random source, a cryptographic key, and a defined table and location in memory.

79. (Previously Presented) The system as in Claim 77, wherein the transformation controller determines an intermixture of the subtask functions of the plurality of software logic modules into the single logic program to provide the combined functionality.

80. (Previously Presented) The system as in Claim 79, wherein the intermixture of the subtask functions of the plurality of software logic modules into the single logic program to provide the combined functionality can be provided in a defined plurality of different ways; and

wherein each of the defined plurality of different ways provides a different version of the single logic program providing the combined functionality.

81. (Previously Presented) The system as in Claim 80, wherein the intermixture of the subtask functions of the plurality of software logic modules into the single logic program to provide the combined functionality is further comprised of at least one of:

obfuscation, encryption, replication, adding dummy code, addition of redundant control, renaming of variables, splitting a procedure into multiple sub-procedure, dictionary transformation, compilation, interpretation, cryptographic transformation, digital signing, and scrambling.

82. (Previously Presented) The system as in Claim 81, wherein the replication is comprised of repetitions of the software logic modules into an oversize program comprising the single logic program embedded therein.

83. (Previously Presented) The system as in Claim 82, wherein each of the repetitions is made active separately to define an active single program within the oversize program which acts as the single logic program.

84. (Previously Presented) The system as in Claim 64, wherein the transformation controller further generates external software modules for linked operation with the single logic program as required for the combined functionality.

85. (Previously Presented) The system as in Claim 84, further comprising:

means for transmitting the external software modules to a separate computing subsystems, and

wherein the external software modules are executed in at least one of the separate computing subsystems to provide at least one of: update information and renewable information coupled to the single logic program.

86. (Original) The system as in Claim 85, wherein the means for transmitting utilizes at least one of: encryption, authentication, and digital signing.

87. (Original) The system as in Claim 85, wherein the update information is at least one of: change data, change executable code, change pattern, change order and pseudo-change of dummy code.

88. (Original) The system as in Claim 85, wherein the renewable information is at least one of:

renewable content processing, time signal, a UTC time signal, a digitally signed time signal, a digital cache for transmission of trusted content processing, and a cryptographic key for marking trusted content processing.

89. (Original) The system as in Claim 64, further comprising:
means for transmitting the single logic program to a primary computing system.

90. (Original) The system as in Claim 89, wherein the means for transmitting utilizes at least one of: encryption, authentication, watermarking, and digital signing.

91. (Original) The system as in Claim 64, wherein security verification information is generated by the transformation controller, for utilization by separate security tag verification logic in a separate subsystem which validates the security tags.

92. (Original) The system as in Claim 64, wherein one of the software logic modules provides security services.

93. (Original) The system as in Claim 92, wherein the security services provide for at least one of:

user authentication, user sign-on, data packet authentication, user login, applying a user's cryptographic key, applying an organization's cryptographic key, group encryption, watermarking validation, and digital signing.

94. (Original) The system as in Claim 92, wherein the security services further provide for applying cryptographic transformations based on keys belonging to a primary computing system.

95. (Original) The system as in claim 94, wherein the primary computing system provides for execution of the single logic program.

96. (Currently Amended) A method of providing controlled signaling for validating execution on a remote computing subsystem, the method comprising:

~~providing defined rules defining at least one of: transmission, forwarding, and operation on a first computing subsystem;~~

receiving streaming data packets from a media server, at the first computing subsystem;

processing of the streaming data packets on the first computing subsystem, in accordance with ~~the~~ defined rules in the first computing subsystem;

generating a security tag responsive to ~~validating~~ the processing in accordance with the defined rules in the first computing subsystem;

transmitting the security tag~~nto a communications path from the first computing subsystem; receiving the security tag from the communications path on~~ to a second computing subsystem; and

validating the security tag on the second computing subsystem ~~so as to validate that the processing of the streaming data packets is~~ responsive to determining that the defined rules were unchanged when the security tag was generated in accordance with the

~~defined rules, and that said processing of the streaming data packets took place at the first computing subsystem only upon successful validation~~processing of the streaming data packets on the first computing subsystem.

97. (Canceled)

98. (Canceled)

99. (Previously Presented) The method as in Claim 96, further comprising:

receiving selected parts of the defined rules on the first computing subsystem from a separate rules controller.

100. (Previously Presented) The method as in Claim 96, further comprising:

determining a renewable software module for at least one of: transmission, forwarding, and operation responsive to at least one of the defined rules on the first computing subsystem of at least one of: transmission, forwarding, and operation on the first computing subsystem.

101. (Previously Presented) The method as in Claim 96, wherein the generating the security tag comprises at least one of:

applying a pseudo-random generator, applying a pseudo-random function, applying a cryptographic function, applying an encryption function, applying a scrambling subroutine, applying an authentication function, applying a digital signing function, applying a cryptographic hash function, applying a subroutine, applying a computational logic module, applying a symmetric cryptography function, applying an asymmetric cryptography function, employing a cryptographic key, employing a cryptographic seed, employing an encrypted software, employing an obfuscated software, employing a hidden program, employing logic with a set of parameters, employing a

hardware module, employing a smart card, employing a portable device, and employing a distributed protocol.

102. (Previously Presented) The method as in Claim 96, wherein the communications path couples a network interface with at least one of:

a computer, a wireless device, handheld device, a Wi-Fi device, a device operating in accordance with IEEE 802.11, a device operating in accordance with IEEE 802.15, a device operating in accordance with IEEE 802.16, a 2.5G cellular telephone, a 3G cellular telephone, a 4G cellular telephone, a 5G cellular telephone, a personal computer, a set-top box, a device operating in accordance with UMTS (Universal Mobile Telephone System), and a device operating in accordance to IEEE 802.3 family of standards.

103. (Previously Presented) The system as in claim 96, wherein the validating confirms that the first computing subsystem is operating in compliance with at least one of digital right management rules and content management system rules.

104. (Previously Presented) The system as in claim 96, wherein the communication path is comprised of defined communications of at least one of: a VPN, an ATM, a FR, a CPN, a content delivery network (CDN), an ISP, a shared media, a firewall, a local area network, an Internet, a metropolitan area network, a SAN, a link to application server, a link to web server, a link to data base server, a link to Internet server, a link to network server, a public network, an enterprise network, and a carrier network.

105. (New) The system as in Claim 1, the system further comprising:

wherein a successful validation determines that the respective security tags were properly generated responsive to said processing of streaming data packets, and

wherein a failed validation determines that the respective security tags were improperly generated responsive to said processing of streaming data packets.

106. (New) The method as in Claim 37, wherein the processing provides playing that provides at leas one of: an input to an audio device and an input to a video display.

107. (New) The method as in Claim 37, further comprising: stopping sending of further data to the first computing subsystem responsive to the failed validation.

108. (New) The method as in Claim 52, wherein the processing in the second computing subsystem further provides for protection of the content.

109. (New) The method as in Claim 54, wherein the processing in the first computer subsystem provides at least one of: an input to an audio device and an input to a video display.